TCP Throughput Profiles Using Measurements Over Dedicated Connections

Nageswara S.V. Rao, Qiang Liu,
Satyabrata Sen
Oak Ridge National Laboratory
Oak Ridge, TN 37831, USA
{raons,liuq1,sens}@ornl.gov

Don Towsley
Gayane Vardoyan
University of Massachusetts
Amherst, MA 01003, USA
{towsley,gvardoyan}@cs.umass.edu

Raj Kettimuthu
Ian Foster
Argonne National Laboratory
Argonne, IL 60439, USA
{kettimut,foster}@mcs.anl.gov

ABSTRACT

Wide-area data transfers in high-performance computing infrastructures are increasingly being carried over dynamically provisioned dedicated network connections that provide high capacities with no competing traffic. We present extensive TCP throughput measurements and time traces over a suite of physical and emulated 10 Gbps connections with 0-366 ms round-trip times (RTTs). Contrary to the general expectation, they show significant statistical and temporal variations, in addition to the overall dependencies on the congestion control mechanism, buffer size, and the number of parallel streams. We analyze several throughput profiles that have highly desirable concave regions wherein the throughput decreases slowly with RTTs, in stark contrast to the convex profiles predicted by various TCP analytical models. We present a generic throughput model that abstracts the ramp-up and sustainment phases of TCP flows, which provides insights into qualitative trends observed in measurements across TCP variants: (i) slow-start followed by well-sustained throughput leads to concave regions; (ii) large buffers and multiple parallel streams expand the concave regions in addition to improving the throughput; and (iii) stable throughput dynamics, indicated by a smoother Poincaré map and smaller Lyapunov exponents, lead to wider concave regions. These measurements and analytical results together enable us to select a TCP variant and its parameters for a given connection to achieve high throughput with statistical guarantees.

CCS CONCEPTS

•Networks →Network performance analysis; Transport protocols;

KEYWORDS

Transport protocols, TCP, dedicated connection, throughput profile, monotonicity, concavity, throughput dynamics, Poincaré map, Lyapunov exponent

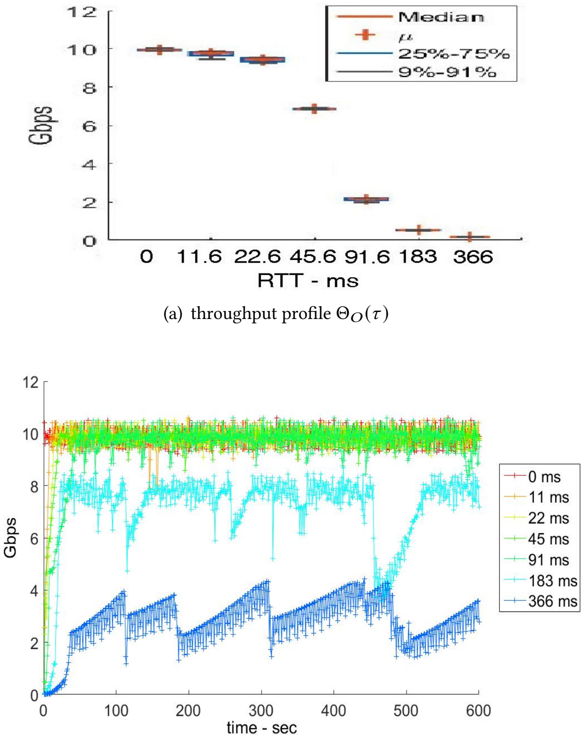
1 INTRODUCTION

Wide-area data transfers over dedicated connections are becoming increasingly important in a variety of scenarios, including multi-site cloud computing server complexes, High-Performance Computing (HPC) workflows, and distributed big data computing facilities [19]. In particular, memory-to-memory transfers are critical to applications such as computations coordinated over cloud servers at geographically dispersed sites, and on-going computations on supercomputers steered by remote analysis and visualization codes. To support such data transfers, network infrastructures, such as Department of Energy's (DOE) ESnet [8] and Google's Software Defined Network (SDN) [11], provide on-demand, dedicated connections. They are expected to provide predictable performance, thereby making it easier to achieve effective and optimized data transfers over them, compared to shared connections.

The dedicated connections play a particularly important role in data transfers between geographically distributed HPC sites, since they are unimpeded by other traffic. Within the DOE HPC infrastructure, special purpose Data Transfer Nodes (DTN) [7] are installed to take advantage of the dedicated OSCARS circuits [17] provisioned over ESnet. Furthermore, Lustre over Ethernet enables file systems to be mounted across long-haul links [2], thereby overcoming the 2.5 ms latency limitation of Infiniband [25]; this approach provides file access over wide area without requiring special transfer tools such as GridFTP [28], XDD [21, 30], or hardware IB range extenders [1, 16, 23]. It is generally expected that the underlying Transmission Control Protocol (TCP) flows over dedicated connections provide peak throughput and stable dynamics that are critical in ensuring predictable transfer performance. However, experimental and analytical studies of such flows are quite limited, since a vast majority of TCP studies focus on shared network environments [27]. More generally, TCP has been widely used for wide-area data transfers, including over dedicated connections. Sustaining high TCP throughput for these transfers requires parameter optimizations specific to dedicated connections. While these optimizations are somewhat easier, they are not simple extensions of the well-studied solutions developed for shared connections.

To gain insights into transport solutions for these dedicated transfers, we systematically collected throughput measurements and time traces for three TCP variants, namely, CUBIC [24], Hamilton TCP (HTCP) [26], and Scalable TCP (SCTP) [12], which are considered to be suitable for high-bandwidth connections. These iperf memory transfer measurements are intended to highlight the performance of TCP, and I/O and file systems are assumed to be of sufficient capacity so as not to impose additional constraints. We use dedicated physical connections and a suite of hardware-emulated

HPDC '17, June 26-30, 2017, Washington, DC, USA
© 2017 ACM. 978-1-4503-4699-3/17/06…$15.00
DOI: http://dx.doi.org/10.1145/3078597.3078615



(a) throughput profile $\Theta_O(\tau)$

(b) time trace $\theta(\tau, t)$

Figure 1: Throughput profile and time traces of Scalable-TCP

10 Gbps connections with 0-366 ms round trip times (RTTs). For a given configuration, hosts, TCP and connection parameters, let $\theta(\tau, t)$ denote the throughput at time t over a connection of RTT τ. Its average over an observation period T_O is called the *throughput profile*:

$$\Theta_O(\tau) = \frac{1}{T_O} \int_0^{T_O} \theta(\tau, t)\, dt$$

Fig. 1 shows representative plots of mean throughput profiles and time traces for a single STCP stream [12].

It is important to note from Fig. 1(a) that the throughput profile $\Theta_O(\tau)$ is *concave* for lower RTTs, but switches to *convex* for larger RTTs. Such dual-mode profiles are observed and analyzed in limited single TCP measurements in [22], but do not seem to be widely known. In this paper, we present extensive TCP measurements with multiple flows and varied buffer sizes, and also present analytical results that relate the extent of concave regions to these parameters and additionally to transport dynamics. From a practical perspective, the concave region is the most desirable characteristic because the throughput decreases slowly as RTT increases, and in particular is higher than the linear interpolation of the end points. In fact, such a profile is in stark contrast to (entirely) convex regions predicted by several TCP models [27], where the throughput decreases faster with RTT and is below the linear interpolation of the end points. Over the past decades, several detailed analytical models have been developed and experimental measurements have been collected for various TCP variants [10, 20, 31]. Based on different loss models, these conventional TCP models provide entirely convex throughput profiles [15, 18, 27], and do not explain this dual-regime profile well. Our measurements demonstrate that both large host (TCP/IP and socket) buffers and more parallel streams expand the concave regions, in addition to improving the throughput. In another direction, throughput time traces exhibit rich dynamics as illustrated in Fig. 1(b), which are much more complex than periodic trajectories predicted by conventional transport models for dedicated connections with no external losses. These dynamics impact the throughput profiles in subtle ways as revealed by our application of Poincaré map and Lyapunov exponent methods from chaos theory [3]: at lower RTTs, higher throughput and smaller variations result in concave regions, and at higher RTTs, lower throughput and larger variations lead to convex regions. Similar and somewhat unexpected complex dynamics have been observed in User Datagram Transport (UDT) transfers [14], which were originally expected to have much smoother dynamics [9].

We propose a generic, coarse throughput model that abstracts the ramp-up (due to slow-start) and sustainment (due to congestion avoidance) phases of TCP and captures the qualitative trends observed in the measurements: (i) exponential ramp-up combined with sustained throughput leads to concave regions, particularly at low RTTs; and (ii) larger buffers and more parallel streams improve the average throughput and also expand the concave region. This model generalizes the single stream model with a fixed buffer size presented in [22]. We compute the Poincaré map of a throughput time trace that specifies the next transfer rate as a function of the current rate, and the Lyapunov exponent that specifies its rate of change. The Poincaré map of an ideal periodic TCP trajectory, predicted by existing models [27] for dedicated connections, is a simple 1-D curve [20]; but, several computed maps form scattered 2-D clusters with positive Lyapunov exponents. Such a 1-D map represents stable dynamics, but in the scattered map cases the nearby throughput values may widely diverge in the next step, indicating much richer dynamics [3]. The Poincaré map may determine critical properties of throughput profiles, and we show that stable throughput dynamics, indicated by a qualitatively compact map and small Lyapunov exponents, expand the concave region.

In addition to providing useful insights, these measurements combined with analytical results provide us practical transport solutions. A TCP variant and its parameters can be chosen using pre-computed throughput profiles to achieve high throughput for a given connection using its RTT, which can be incorporated into HPC wide-area infrastructures [19] and HPC I/O frameworks [5, 13]. Furthermore, throughput under this configurations can be estimated by interpolating the measurements with certain statistical guarantees without the knowledge of underlying joint error distributions of connections and host systems.

Various measurements and experimental configurations are described in Section 2. A generic throughput model is presented in Section 3, including illustrations of monotonicity and dual-regime concavity/convexity. The time traces and stability properties are discussed in Section 4. A method for selecting a transport method for a given connection and statistical guarantees of its throughput estimates are presented in Section 5. Conclusions are presented in Section 6.

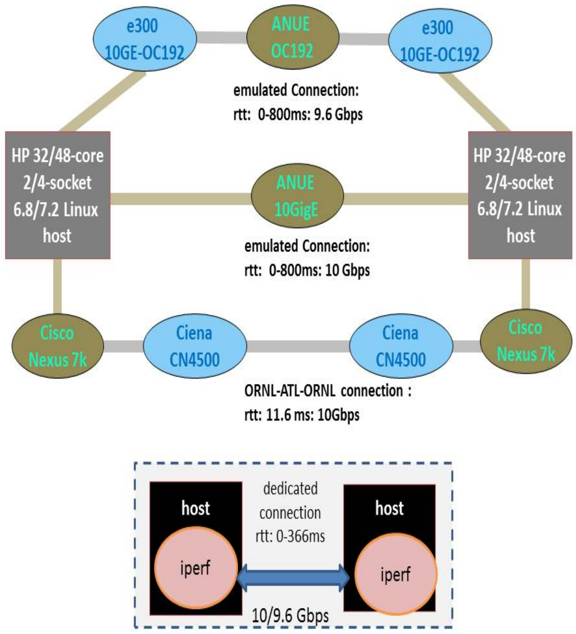


Figure 2: Testbed connections

2 THROUGHPUT MEASUREMENTS AND PROFILES

We collected extensive TCP throughput measurements over the past two years, using three TCP variants, three buffer sizes, 1-10 parallel streams over connections of seven different lengths and two modalities, as shown in Table 1. Their throughput profiles share certain common qualitative properties in terms of monotonicity and concave/convex regions, which we summarize in this section; we do not, however, attempt to provide a comprehensive analysis of these large data sets.

2.1 Measurement Testbed

Measurements are collected over our testbed with four 32-core HP Linux workstations Feynman1 ($f1$) through Feynman4 ($f4$) with Linux kernel 2.6 ($f1$,$f2$) and Linux kernel 3.10 ($f3$,$f4$). Hosts with identical configurations are connected in pairs over a back-to-back fiber connection with negligible 0.01 ms RTT and a physical 10GigE connection with 11.6 ms RTT via Cisco and Ciena devices, as shown in Fig. 2. Two different physical modalities are represented by the 10GigE and SONET/OC192 connections. For the latter, 10GigE NICs are connected to a Force10 E300 switch that converts between 10GigE and SONET frames, and the OC192 ANUE emulator is in turn connected to WAN ports of E300, as shown in the top connection in Fig. 2. We use suites of emulated 10GigE and SONET/OC192 connections via ANUE devices with RTTs $\tau \in \{0.4, 11.8, 22.6, 45.6, 91.6, 183, 366\}$ ms. The lower RTTs represent cross-country connections, for example, between facilities across the US; higher RTTs 93.6 and 183 ms represent inter-continental connections, for example, between US, Europe, and Asia; and the 366 ms RTT represents a connection spanning the globe.

TCP memory-to-memory throughput measurements and parameter traces are collected for three TCP congestion control modules using the *iperf* and *tcpprobe* kernel module. The number of parallel streams is varied from one to ten for each configuration, and throughput measurements are repeated ten times. The configuration for iperf includes transfer sizes of default (around 1 GB), 20 GB, 50 GB, and 100 GB. TCP buffer sizes are default, normal (recommended values for 200 ms RTT), and large (the largest size allowed by the kernel); and the socket buffer parameter for iperf is 2 GB. The net effects of these settings result in the allocation of 250 KB, 250 MB and 1 GB socket buffer sizes, respectively.

2.2 TCP Measurements

We compute the mean throughput profile by taking the mean of the average throughput rates from repeated transfer experiments conducted at specific τ values and numbers of parallel streams. The results are collectively shown in Figs. 3-6 for select configurations. From these plots, the overall trend can be easily observed: the mean throughput generally decreases with increasing RTTs, and increases with more streams.

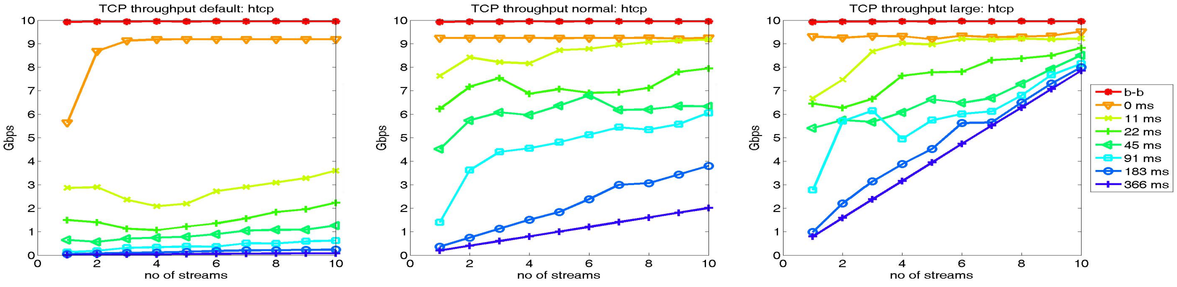
In Fig. 3, mean throughput rates under three buffer sizes, namely, default, normal, and large, are plotted for HTCP with $f1\_sonet\_f2$ configuration. A larger buffer size significantly improves the mean throughput, especially for longer connections; for instance, throughput of 10 streams for 366 ms RTT improves from 100 Mbps to nearly 8 Gbps as the buffer size increases. The throughputs of CUBIC and STCP are very close to their HTCP counterparts. Unless otherwise specified, subsequent discussions primarily address performance with large buffers.

Figs. 4 and 5 show the mean throughput of STCP and CUBIC, respectively. For STCP, compared to SONET, the 10GigE ($f1\_10gige\_f2$) link improves the mean throughput for low-to-mid RTTs, especially for higher stream counts; the difference is less pronounced for CUBIC in the same RTT range, with little to no improvement for both versions under higher RTTs. On the other hand, transfer rates between hosts with Linux kernel 3.10 ($f3\_sonet\_f4$) are minimally affected by connection modality (OC192 or 10GigE) under most RTTs and somewhat worsened under 366 ms RTT using STCP. For CUBIC, changes are also seen mostly for high RTTs: while the cases with lower stream counts seem to benefit the most from the new kernel, more streams result in degraded performance just as with STCP. In what follows, we will focus on the $f1$-$f2$ configurations.

So far, the TCP transfer size has been set to "default". When a fixed larger transfer size is imposed, the performances are quite different. In Fig. 6, each graph illustrates throughput as a function of number of streams and RTTs for different transfer sizes. Throughputs generally increase as a function of the transfer size, especially for larger RTTs. Recall that the average throughput is a weighted average between the ramp-up and sustainment phases; an increased transfer size effectively prolongs the sustainment stage, and thereby improves overall throughput. It is also interesting to note that with large transfer sizes, the throughput profiles (with increasing numbers of streams) become flatter for most RTTs, indicating the reduced effect of adopting multiple streams.

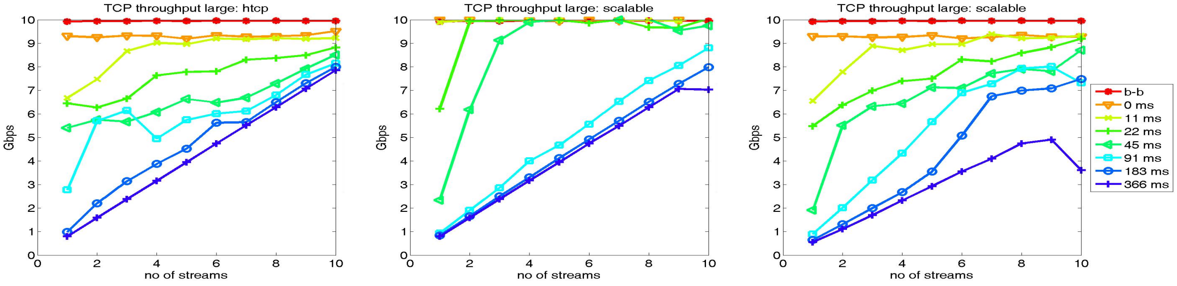
Table 1: Configurations

option	parameter range
host OS	feynman1-2 (Linux kernel 2.6, CentOS 6.8), feynman3-4 (Linux kernel 3.10, CentOS 7.2)
congestion control	CUBIC, HTCP, STCP
buffer size	default (244 KB), normal (256 MB), large (1 GB)
transfer size	default (≈ 1 GB), 20 GB, 50 GB, 100 GB
no. streams	1–10
connection	SONET-OC192 (9.6 Gbps), 10GigE (10 Gbps)
RTT	0.4, 11.8, 22.6, 45.6, 91.6, 183, 366 ms



(a) default (b) normal (c) large

Figure 3: Throughput with variable RTTs, number of streams, and buffer sizes for HTCP with $f1\_sonet\_f2$ configuration



(a) $f1\_sonet\_f2$ (b) $f1\_10gige\_f2$ (c) $f3\_sonet\_f4$

Figure 4: Throughput with variable RTTs, number of streams, and configurations for STCP with large buffers

2.3 Profiles and Transitions

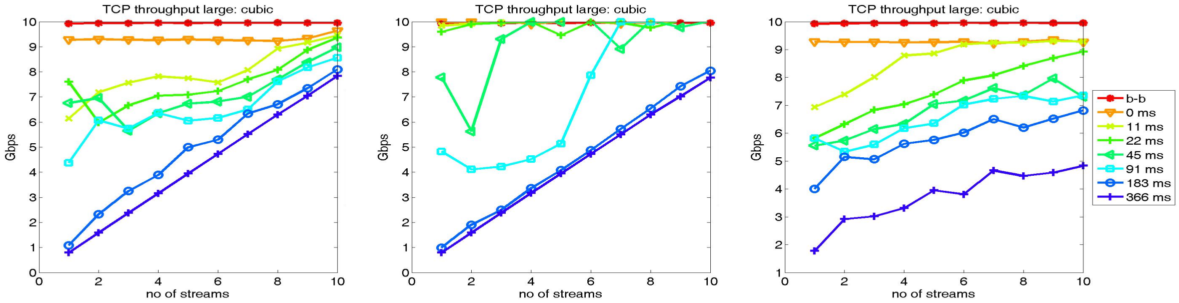
Throughput box plots for CUBIC with large buffers in Fig. 7 show that compared to SONET, 10GigE throughput rates in general exhibit less variation. More importantly, in both cases, using more streams not only increases the throughput, but also extends the concave region, as the convex region around larger RTTs with a single stream largely disappears with 10 streams. In addition, the buffer size also has a similar effect, namely, increased mean throughput and extended concave region. As seen from Fig. 8, for CUBIC with 10 streams over SONET, using the default buffer size results in an entirely convex profile; with the normal buffer size, a concave region (leading up to 91.6 ms) is followed by a convex region; finally, a large buffer extends the concave region all the way beyond 183 ms.

We compute the *transition-RTT* τ_T between concave and convex regions by regression fitting a pair of sigmoid functions, as illustrated in Fig. 9. Using the flipped sigmoid function $g_{a_1,\tau_1}(\tau) = 1 - \frac{1}{1+e^{-a_1(\tau-\tau_1)}}$, we fit concave-convex switch regression function

$$f_{\Theta_O}(\tau) = g_{a_1,\tau_1}(\tau) I(\tau \leq \tau_T) + g_{a_2,\tau_2}(\tau) I(\tau \geq \tau_T)$$

where $I(\cdot)$ is the indicator function, $g_{a_1,\tau_1}(\tau)$ is the concave fit, and $g_{a_2,\tau_2}(\tau)$ is the convex fit. The concave and convex portions of the regression model are ensured by constraining $\tau_2 \leq \tau_T \leq \tau_1$. We calculate the parameters a_1, τ_1, a_2, τ_2, and the transition-RTT τ_T by minimizing the sum-squared error (SSE) between the measured throughput values and the fitted model, defined as

$$\text{SSE} = \sum_{\tau \leq \tau_T} \left(\widetilde{\Theta}_O(\tau) - g_{a_1,\tau_1}(\tau)\right)^2 + \sum_{\tau \geq \tau_T} \left(\widetilde{\Theta}_O(\tau) - g_{a_2,\tau_2}(\tau)\right)^2,$$

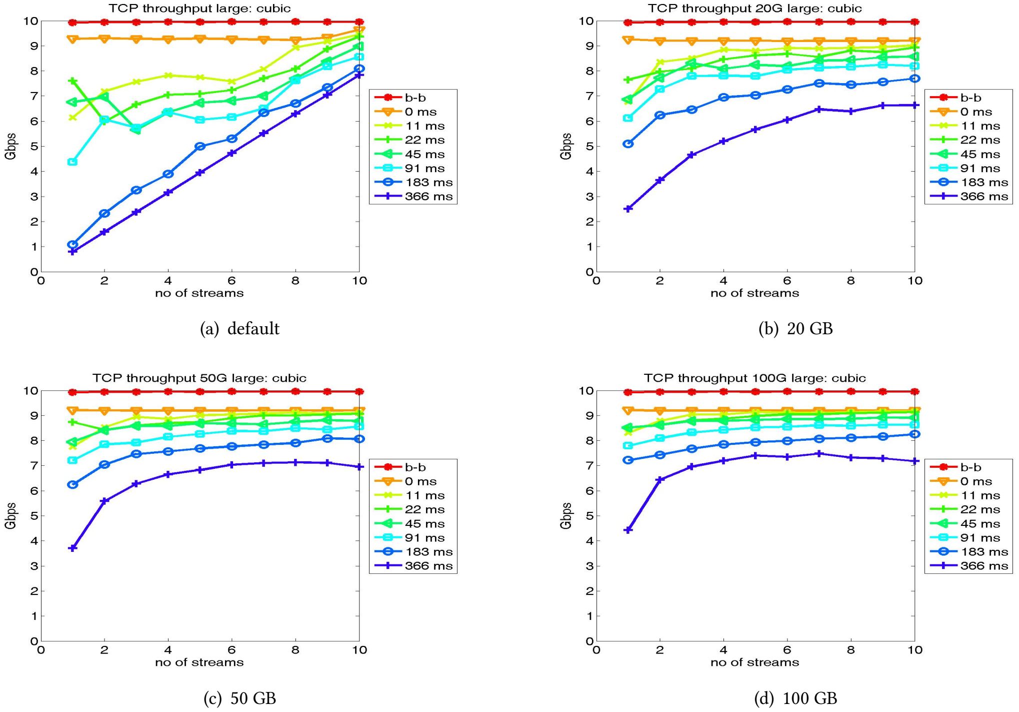


(a) $f1\_sonet\_f2$

(b) $f1\_10gige\_f2$

(c) $f3\_sonet\_f4$

Figure 5: Throughput with variable RTTs, number of streams, and configurations for CUBIC with large buffers



(a) default

(b) 20 GB

(c) 50 GB

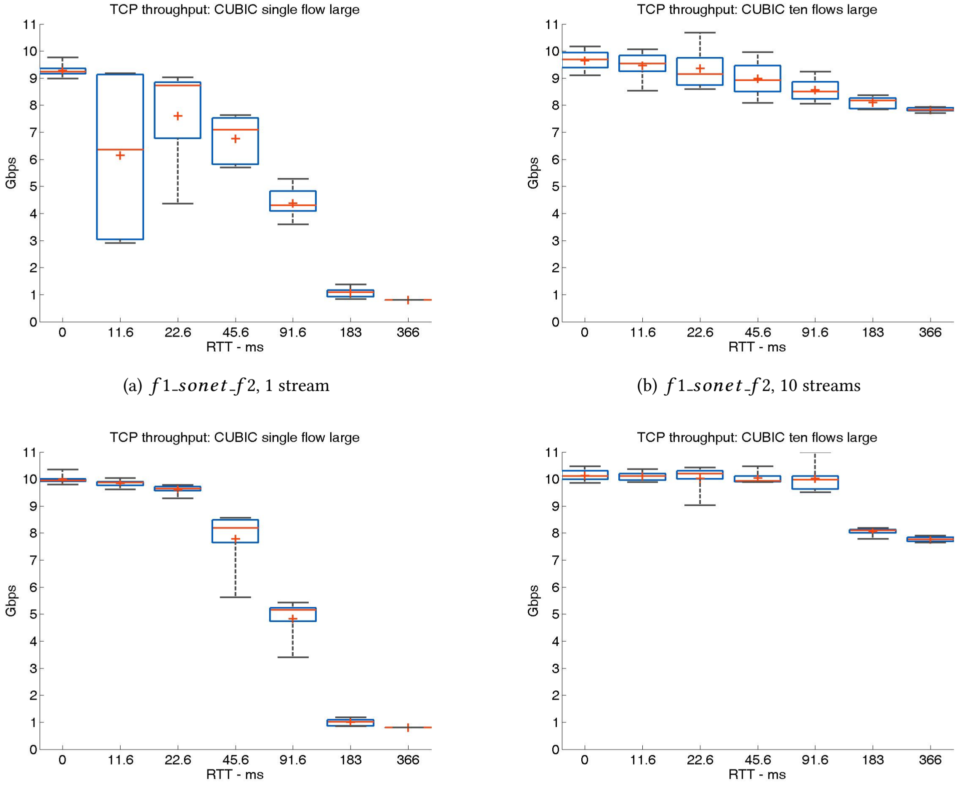
(d) 100 GB

Figure 6: Throughput with variable RTTs, number of streams, and transfer sizes for CUBIC with large buffers and $f1\_sonet\_f2$ configuration

where $\widetilde{\Theta}_O(\tau) \in (0,1)$ is the scaled version of the measured throughput values.

In Fig. 9, we demonstrate the fitted sigmoid models along with the measured throughput profiles for three different buffer sizes, with a single CUBIC stream over 10GigE. As mentioned earlier, the profile is entirely convex at the default buffer size, and consequently there is only a convex portion to the sigmoid fit. For normal and large buffer sizes, both the concave and convex sigmoid fits are present, respectively are shown with solid-blue and dashed-black curves. It is clear that τ_T increases, hence the concave region extends, as the buffer size is increased.

We repeat the regression model fit for 1-10 parallel streams and three congestion control modules. The overall variations of the estimated transition-RTT values w.r.t. number of parallel streams, buffer sizes, and TCP congestion control modules are shown in Fig. 10. For CUBIC, while using the default buffer size, the transition-RTT values increase from 0.4 ms for 1-3 parallel streams to 11.8 ms for 4 or more parallel streams. When the normal buffer size is used, the transition-RTT values remain consistently higher (at 45.6 ms, except for 2 steams) than those with the default buffer size, and further increase to 91.6 ms for 10 parallel streams. The τ_T estimates with the large buffer size show even larger values than both the default and normal buffer sizes; for example, 91.6 ms for 1-6 parallel



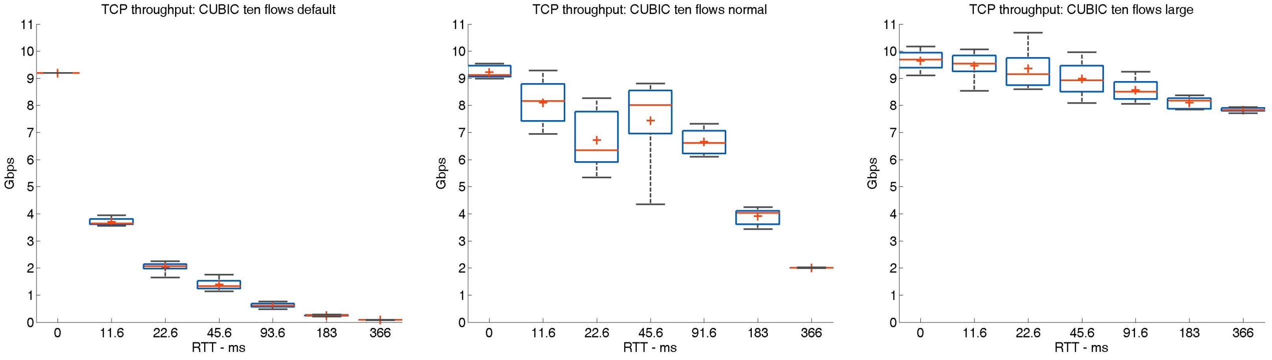
(a) $f1\_sonet\_f2$, 1 stream

(b) $f1\_sonet\_f2$, 10 streams

(c) $f1\_10gige\_f2$, 1 stream

(d) $f1\_10gige\_f2$, 10 streams

Figure 7: Throughput box plots with variable RTTs, stream counts, and configurations for CUBIC with large buffers



(a) default

(b) normal

(c) large

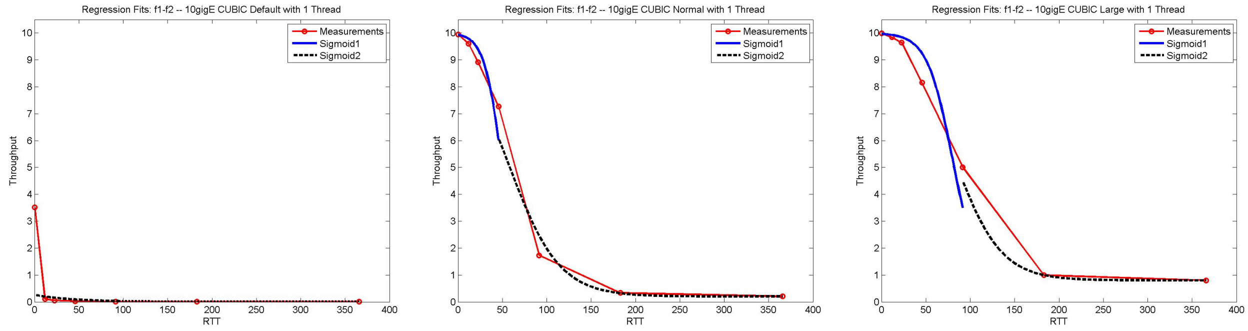
Figure 8: Throughput box plots with variable RTTs and buffer sizes for CUBIC with 10 streams and $f1\_sonet\_f2$ configuration

streams (except 2 streams) and 183 ms for 7 or more parallel streams. Similar increasing trends of the estimated τ_T values are also noted for HTCP and STCP, and thereby corroborate our inference that more streams and larger buffer sizes extend the concave region in addition to improving the throughput.

3 THROUGHPUT MODEL

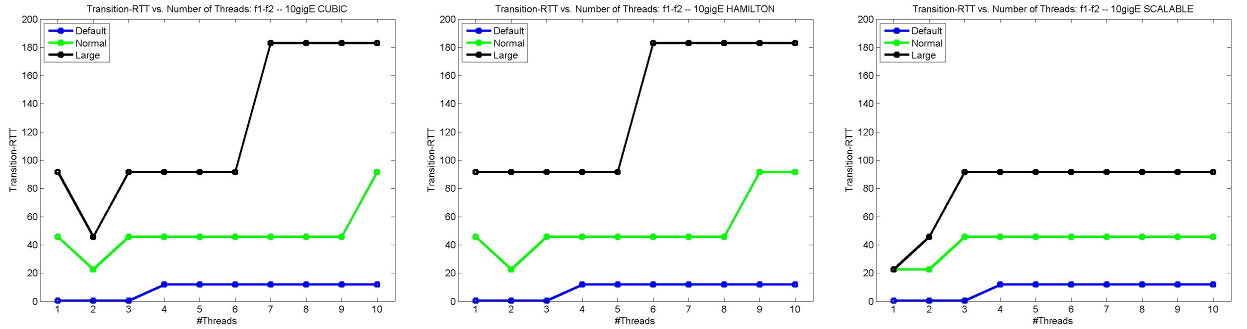
We now present a generic throughput model[1] to explain the overall qualitative behavior observed in TCP measurements in previous section. A similar model has been presented for UDT in [14].

[1] This is a coarse or approximate model mainly aimed at explaining the concave-convex transitions in measured throughput profiles, and does not capture the complex, disparate congestion control details of TCP variants.



(a) default (b) normal (c) large

Figure 9: Sigmoid regression fits of throughput profiles with various buffer sizes for single stream CUBIC and $f1\_10gige\_f2$ configuration



(a) CUBIC (b) HTCP (c) STCP

Figure 10: Transition-RTT estimates with 1-10 streams and various buffer sizes for CUBIC, HTCP, and STCP in $f1\_10gige\_f2$ configuration

3.1 Basic Throughput Model

The throughput profiles are determined by: (i) protocol parameters including version $V = C, H, S$, representing CUBIC, HTCP, and STCP, respectively, number of parallel streams n, and socket buffer size B allocated during measurements that is a result of cumulative effects of TCP/IP host and socket parameters at sending and receiving hosts; (ii) connection RTT, modality, and capacity, for example, 10 Gbps for 10GigE and 9.6 Gbps for SONET OC192; and (iii) settings of the measurement tool such as the duration and data transfer size for iperf. Let $\theta_V^{B,n}(\tau, t)$ denote the aggregate throughput at time t over a connection of RTT τ. These parameters may be explicitly used to denote the throughput profile as

$$\Theta_{V,O}^{B,n}(\tau) = \frac{1}{T_O} \int_0^{T_O} \theta_{V,O}^{B,n}(\tau, t)dt,$$

and selectively suppressed when evident from the context.

The throughput dynamics of a transport with fixed parameters over a connection with RTT τ and capacity C are characterized by two phases:

(a) **Ramp-Up Phase:** In the ramp-up phase, $\theta(t)$ increases for a duration of T_R until it reaches a peak $C_\tau^{B,n} \leq C$, which depends on B and n, and then switches to a sustained throughput phase. The ramp-up phase corresponds to the slow-start of TCP, in which $\theta(t)$ increases exponentially as the congestion window *cwnd* $w(t)$ grows. The specifics of slow-start may vary among different TCP variants and their implementations. The average throughput in this phase is

$$\bar{\theta}_R(\tau) = \frac{1}{T_R} \int_0^{T_R} \theta(\tau, t)dt.$$

(b) **Sustained Throughput Phase:** Once throughput reaches the peak $C_\tau^{B,n}$, it is "sustained" using a mechanism which processes the acknowledgments, and infers and responds to losses. For TCP, this is the congestion avoidance phase that follows the slow-start, wherein $w(t)$ is incremented somewhat slowly by an amount specific to the TCP version. The average throughput in this region is

$$\bar{\theta}_S(\tau) = \frac{1}{T_S} \int_{T_R}^{T_R+T_S} \theta(\tau, t)dt.$$

In general, the average $\bar{\theta}_S$ lies below $C_\tau^{B,n} \leq C$ due to variations in $\theta(t)$ as shown in Fig. 1(b).

The average throughput is

$$\Theta_O(\tau) = \frac{T_R}{T_O}\bar{\theta}_R(\tau) + \frac{T_S}{T_O}\bar{\theta}_S(\tau) = \bar{\theta}_S(\tau) - f_R\left(\bar{\theta}_S(\tau) - \bar{\theta}_R(\tau)\right),$$

where $T_O = T_R + T_S$ and $f_R = T_R/T_O$. For a large observation period T_O with a fast ramp-up, typical in small τ settings, the qualitative properties of $\theta_S(\tau)$ directly carry over to $\Theta_O(\tau)$. For large τ, however, the ramp-up period can take significantly longer, for example, 10 seconds for 366 ms RTT (Fig. 1(b)), and the difference term $\bar{\theta}_S - \bar{\theta}_R$ modulates the behavior of $\theta_S(\tau)$ in determining that of $\Theta_O(\tau)$.

3.2 TCP Model for Dedicated Connections

A function $f(\tau)$ is *concave* [6] in interval I if for any $\tau_1 < \tau_2 \in I$, the following condition is satisfied: for $x \in [0, 1]$

$$f\left(x\tau_1 + (1 - x)\tau_2\right) \geq xf(\tau_1) + (1 - x)f(\tau_2).$$

It is *convex* if $\geq$ in the above condition is replaced by $\leq$. A function is concave if and only if $\frac{df}{d\tau}$ is a non-increasing function of τ or equivalently $\frac{d^2f}{d\tau^2} \leq 0$. Using a simplified model, TCP memory-to-memory transfers for the special case of unlimited host buffers ($B = \infty$) and a single stream ($n = 1$) have been shown to have two basic regions [22]:

(a) *Concave Region:* For smaller RTTs, as the congestion window $w(t)$ crosses the slow-start threshold W_{SS}, $\theta(t)$ switches from exponentially increasing to a constant C (lower RTT plots in Fig. 1(b)). This behavior leads to the concave profile that we observed in measurements, as will be shown in the next section.

(b) *Convex Region:* For larger RTTs, $w(t)$ crosses W_{SS} before reaching $C\tau$, and its slower growth in the congestion avoidance model leads to losses (higher RTT plots in Fig. 1(b)) and a convex profile.

In most cases shown in the previous section, the profile is concave when RTT is small, and at transition RTT τ_T it becomes and continues to be convex as RTT increases. This behavior is in part a result of various host buffers being sufficiently large to fill up the connection to the near capacity C combined with the fast response of TCP congestion control at lower RTT. In this region we term the protocol to be *peaking at zero* (PAZ) since $\lim_{\tau \to 0} \Theta_O(\tau) \approx C$, as illustrated in Figs. 3, 4, and 6, particularly over the back-to-back connection.

Traditional TCP models, driven primarily by losses, lead to throughput profiles in the generic form $\hat{\mathcal{T}}(\tau) = a + b/\tau^c$, for suitable constants a, b, and $c \geq 1$ [27]. These convex profiles (since $\frac{d\mathcal{T}}{d\tau} = -b/\tau^2$ increases with τ) are typical of large transfers and longer RTTs, and do not adequately account for transfers that lead to concave portions in the observed profiles.

3.3 Monotonicity of Throughput Profile

In the average throughput $\Theta_O(\tau) = \bar{\theta}_S - f_R\left(\bar{\theta}_S - \bar{\theta}_R\right)$, the ramp-up duration increases with τ and hence f_R increases. This condition is sufficient to show that in PAZ cases, the throughput profile decreases with RTT. Consider a best-case scenario of a sustainment phase such that $\theta_S(t) \approx C$, that is, B and n are sufficiently large, and TCP variant V is ideally responsive to RTT τ to completely fill the connection capacity. Then $\Theta_O(\tau)$ decreases with RTT since $\bar{\theta}_R \leq C$, and $\bar{\theta}_S - \bar{\theta}_R > 0$. This property carries over to the more general case where $\theta_S(t)$ decreases with RTT, since the recovery time from losses increases and results in a lower $\bar{\theta}_S$. In general $\theta_S(\tau)$ decreases with τ in part as a result of TCP's self-clocking behavior that makes its response slower, which in effect enhances the monotonic decrease of $\Theta_O(\tau)$. There are two different ways such a decrease manifests, as will be shown in the next section: if B or n is sufficiently large, the decrease is slower and leads to the concave profile region, otherwise a faster decrease leads to the convex profile region. However, if throughput falls much below C after the ramp-up and has significant random variations, it is quite possible for $\Theta_O(\tau)$ to increase with respect to τ in certain albeit small regions (Fig. 8(b)), but our measurements show mostly decreasing profiles.

3.4 Multiple Flows and Large Buffers

We now consider a base case where the slow-start phase is followed by well-sustained throughput such that $\theta_S(\tau) \approx C$. For an exponential increase during the slow start of TCP, the throughput reaches C in $n_R = \log C$ steps, and the total data sent during T_R period is $2C$. Thus, we have $T_R = \tau \log C$, and $\bar{\theta}_R = \frac{2C}{\tau \log C}$, and

$$\Theta_O = \frac{2C}{T_O} + C\left(1 - \frac{\tau \log C}{T_O}\right).$$

Then, we have $\frac{d\Theta_O}{d\tau} = -C \log C/T_O$, a non-increasing function of τ, which shows concavity when throughput is maintained close to C. For a small τ, as shown in Fig. 1(b), TCP traces indicate smaller variations, which leads to $\bar{\theta}_S$ values around the peak C. However, for a large τ, deeper decreases in the traces lead to a lower $\bar{\theta}_S$, and in turn convex profiles as shown in Fig. 1(b).

Consider that throughput increases faster than exponential such that $T_R = \tau^{1+\epsilon} \log C$, for $\epsilon > 0$ as in the case of n TCP streams. Then,

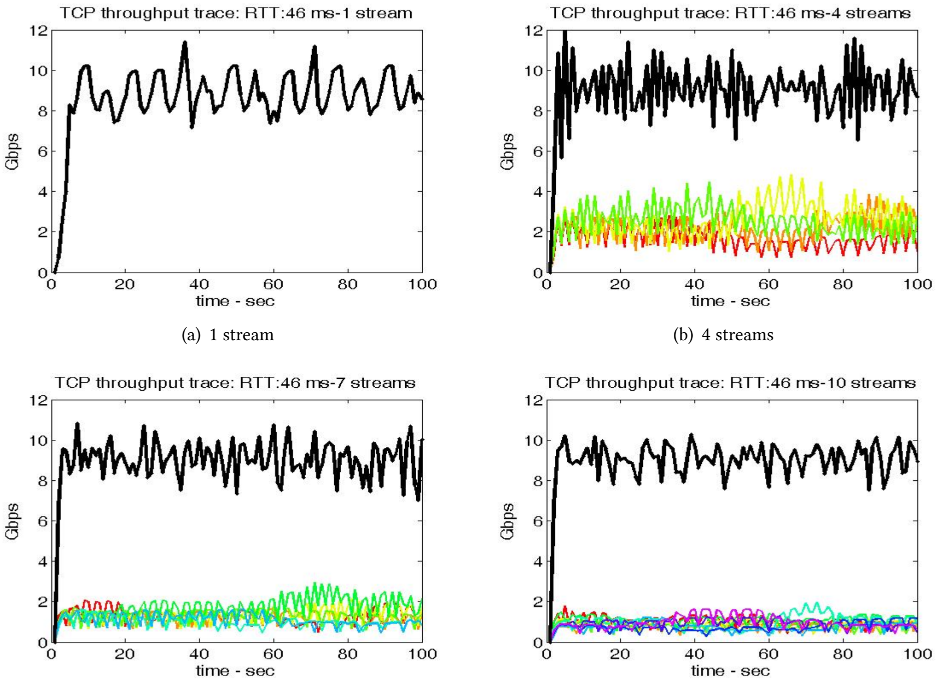
$$\Theta^n_O(\tau) = \frac{2C}{T_O} + C\left(1 - \frac{\tau^{1+\epsilon} \log C}{T_O}\right)$$

and

$$\frac{d\Theta^n_O}{d\tau} = -C \log C/T_O(1 + \epsilon)\tau^{\epsilon},$$

which is a decreasing function of τ that leads to a concave $\Theta_O(\tau)$. On the other hand, for a slower-than-exponential increase, consider $T_R = \tau^{-\epsilon} \log C$, for a small $\epsilon > 0$. We have $\frac{d\Theta}{d\tau} = -C \log C/T_O(1 - \epsilon)\tau^{-\epsilon}$, which is an increasing function of τ that leads to a convex $\Theta_O(\tau)$. Thus, the exponential increase ramp-up followed by sustained throughput, $\theta_S(t) \approx C$, represents a transition point for the profiles: either slower ramp-up or unsustained peak can result in convex profiles.

For buffer sizes $B_1 < B_2$, we have $\theta^{B_1}_S(\tau) \leq \theta^{B_2}_S(\tau)$, which leads to $\Theta^{W_1}_S(\tau) \leq \Theta^{B_2}_S(\tau)$. Thus, as the buffer size increases, the protocol operates closer to the PAZ region, which combined with suitably sustained throughput leads to a concave region. Then, the monotonicity of profiles implies $\tau^{B_1}_T \leq \tau^{B_2}_T$; that is, a larger buffer size results in an expanded concavity region, as indicated in Fig. 8(c).



(a) 1 stream

(b) 4 streams

(c) 7 streams

(d) 10 streams

Figure 11: Throughput traces for CUBIC with $f1\_sonet\_f2$ configuration, large buffers, 45.6 ms RTT, and 1,4,7 and 10 streams

3.5 Concave Region Boundaries

The concave region is characterized by non-increasing derivative

$$\begin{aligned}\frac{d\Theta_O(\tau)}{d\tau} &= \frac{d\bar{\theta}_S(\tau)}{d\tau} - f_R\left(\frac{d\bar{\theta}_S(\tau)}{d\tau} - \frac{d\bar{\theta}_R(\tau)}{d\tau}\right)\\ &\quad -\frac{df_R}{d\tau}\left(\bar{\theta}_S(\tau) - \bar{\theta}_R(\tau)\right),\end{aligned}$$

which is determined by $\bar{\theta}_S(\tau) - \bar{\theta}_R(\tau)$ and its derivative. Now f_R described in the previous section is almost linear in τ, and hence its derivative is constant. Consider that the derivatives are much larger such that effects of $\bar{\theta}_S(\tau) - \bar{\theta}_R(\tau)$ are not dominant; also, the throughput in the sustainment phase decreases faster than in slow-start, i.e., $\left|\frac{d\theta_S}{d\tau}\right| \geq \left|\frac{d\theta_R}{d\tau}\right|$. Then, the term $\left(\frac{d\bar{\theta}_S(\tau)}{d\tau} - \frac{d\bar{\theta}_R(\tau)}{d\tau}\right)$ is negative, which in turn makes the second term of $\frac{d\Theta_O(\tau)}{d\tau}$ a non-decreasing function of τ (since f_R is an increasing function of τ). Then, if $\left|\frac{d\theta_S}{d\tau}\right|$ is much higher, then the increase due to the second term amplified by f_R offsets the decrease in the first term, thereby leading to an overall convex profile. Increasingly large variations in the time traces lead to corresponding increases in $\left|\frac{d\theta_S}{d\tau}\right|$, and the profile consequently transitions to a convex region.

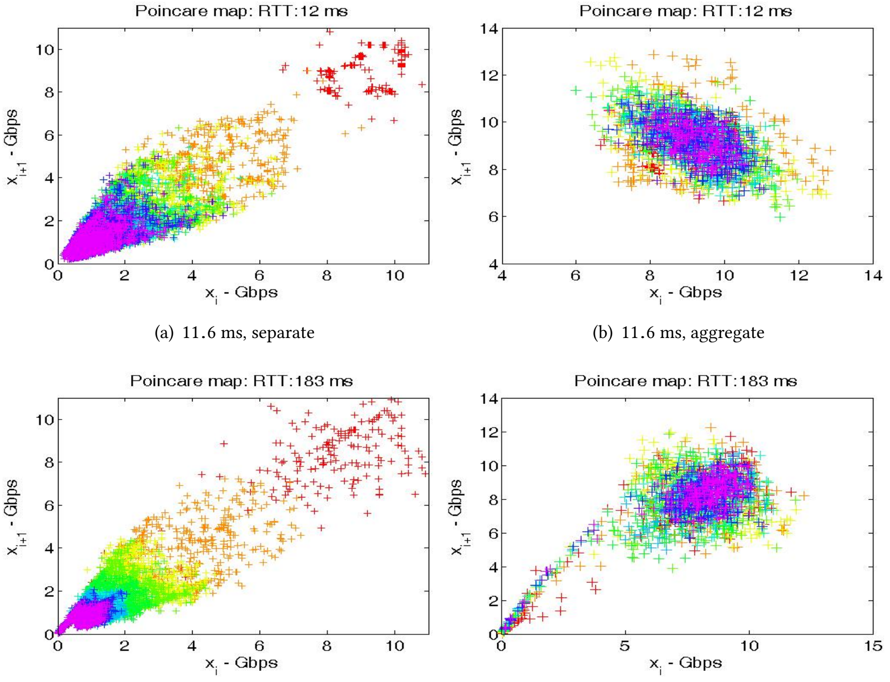
4 DYNAMICS OF THROUGHPUT TRACES

Time traces of throughput measurements provide more detailed information about the transfer processes than mean throughput profiles. Our main objective is to characterize their dynamics and stability as related to throughput profiles. We use tools from chaos theory, the Poincaré maps and the Lyapunov exponents (Section 4.1), to relate stability properties to throughput profiles, in particular, their concave and convex regions at peak throughputs (Section 4.2).

We generate throughput traces by sampling at one-second intervals for a total duration of 100 seconds. However, in contrast to measurements in Section 2, the transfer size here is not fixed, and a higher average throughput indicates a larger transfer size. Fig. 11 shows typical traces of CUBIC throughput over 45.6 ms RTT SONET, with large buffers and variable numbers of streams. In these plots, the thick black curves describe the aggregate transfer rates, whereas different colored curves are transfer rates for individual streams. As evident from these plots, while the per-stream transfer rate decreases with more streams, the aggregate rates across the cases appear to hover around 9 Gbps; in this case, the transfer size is around 100 GB for most cases, and the average throughput rates are more or less consistent with the mean profile shown in Fig. 6(d).

4.1 Poincaré Map and Lyapunov Exponent

A *Poincaré Map* $M : \Re_d \longmapsto \Re_d$ corresponds to a real-vector state $X_i \in \Re_d$ updated at each time step i such that $X_{i+1} = M(X_i)$ [3], and in our context, the sequence $X_0, X_1, ..., X_t$ corresponds to a throughput trace. Then, the Poincaré map generated from a throughput trace provides critical insights into the dynamics of the underlying transport method. Ideal TCP periodic traces lead to maps that form 1-D curves [20], and ideal UDT traces form 1-D monotone curves [14]. In general, the complex geometry of these maps, such as 2-D clusters, represents complex dynamics, since similar throughput values in the current step evolve into wildly



(a) 11.6 ms, separate

(b) 11.6 ms, aggregate

(c) 183 ms, separate

(d) 183 ms, aggregate

Figure 12: Poincaré maps for CUBIC with $f1\_sonet\_f2$ configuration, large buffers, and 11.6 ms vs. 183 ms RTTs

different ones in the next step. The trace of a Poincaré map M can be characterized by its *Lyapunov Exponent* $\mathcal{L}_M = \ln\left|\frac{dM}{dX}\right|$, which describes the separation of trajectories that originate from nearby states. The negative Lyapunov exponents correspond to stable dynamics and positive values represent exponentially divergence traces and possibly chaotic dynamics [3].

Fig. 12 displays the Poincaré maps for CUBIC with variable stream counts, large buffers, and 11.6 ms and 183 ms RTTs over SONET connections. In the plots labeled as "separate", per-stream Poincaré maps are plotted; more specifically, starting from the top right corner, each color represents an increasing stream count, from 1 all the way to 10 streams on the bottom left corner. Comparing Figs. 12(a) and (c), we observe that with single stream, the (red) 183 ms trace transfer rates occupy a much wider region than the 11.6 ms ones, indicating the larger variations – and the reduced average throughput – of the former. With 10 streams, though, per-stream transfer rates with 11.6 ms RTT become much larger than those in the 183 ms cases, as seen from the wider (purple) area.

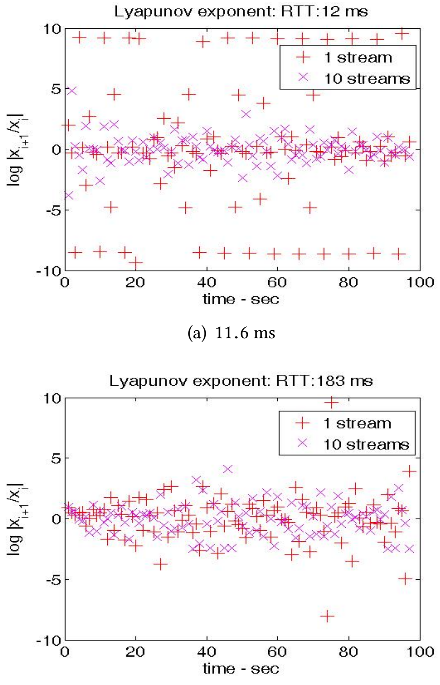
On the other hand, in the aggregate transfer rate Poincaré maps in Figs. 12(b) and (d), the points are superimposed on top of one another with varying flow counts, forming a cluster that describes the sustainment stage. In particular, the 183 ms RTT case demonstrates the effect of a longer ramp-up stage by the points from the origin leading up to the cluster, absent in the 11.6 ms RTT case. Interestingly, the "tilts" of the two clusters appear different: whereas the 183 ms RTT cluster in Fig. 12(b) aligns more with the ideal 45° line, the 11.6 ms cluster tilts to the left, indicating a less stable profile of the corresponding time traces (even with overall higher mean throughput rates). This can be further confirmed with the Lyapunov exponent plots shown in Fig. 13, where the points in the 183 ms case are more compact and closer to the zero line as opposed to the 11.6 ms case. In addition, both plots also reveal that using more streams can reduce the instability in aggregate transfer rates by pulling the Lyapunov exponents closer to zero.

4.2 Peak Throughput and Instability

The variations in throughput traces are a result of protocol dynamics, which in turn determine the average throughput. In particular, positive Lyapunov exponents play a critical role in determining the throughput of protocols that operate at peak throughput, since the diverging trajectories can only be below the peak and larger exponents lead to lower throughput. For throughput rate θ_S, let θ_{S-} be the corresponding sending rate in the previous Poincaré iteration, which is given by the inverse of the ideal Poincaré map at θ_S. Let $L(\theta_{S-})$ denote the corresponding Lyapunov exponent. Then the derivative $\frac{\partial \theta_S}{\partial \theta_{S-}} = e^{L(\theta_{S-})}$ could be large for positive Lyapunov exponents, e.g., those shown in Fig. 13. For fixed f_R and θ_R, we have

$$\frac{\partial \Theta_O}{\partial \tau} = (1-f_R)\frac{\partial \bar{\theta}_S}{\partial \tau} = (1-f_R)\frac{\partial \bar{\theta}_S}{\partial \bar{\theta}_{S-}}\frac{\partial \bar{\theta}_{S-}}{\partial \tau},$$

which indicates the amplifying effects of positive $L(\theta_{S-})$. Consider two configurations C_1 and C_2 with Lyapunov exponents L_1 and L_2, respectively, such that $L_1 > L_2$, and $\bar{\theta}_S^1$ and $\bar{\theta}_S^2$ are their average throughput in sustainment phase, respectively. The trajectories of C_1 will have larger deviations than those of C_2, thereby leading to $\bar{\theta}_S^1 \leq \bar{\theta}_S^2$. This phenomenon is observed in Fig. 14 where there is an overall decreasing relationship between the Lyapunov exponent



(a) 11.6 ms

(b) 183 ms

Figure 13: Lyapunov exponents for CUBIC over 11.6 **ms and** 183 **ms RTT SONET and large buffers**

and average throughput. Now by fixing $\bar{\theta}_S$, consider

$$\frac{\partial \Theta_O}{\partial \tau} = -\frac{\partial f_R}{\partial \tau}(\bar{\theta}_S - \bar{\theta}_R).$$

Since $\frac{\partial f_R}{\partial \tau} \geq 0$, the concavity of Θ_O is equivalent to the condition $\bar{\theta}_S - \bar{\theta}_R \geq 0$. Then, for a fixed configuration, the condition $\bar{\theta}_S^1 \leq \bar{\theta}_S^2$ in turn leads to $\{\tau : \bar{\theta}_S^1 \geq \bar{\theta}_R\} \subseteq \{\tau : \bar{\theta}_S^2 \geq \bar{\theta}_R\}$, which shows that configuration C_2 has a broader concavity region. Thus, lower throughput variations are desirable in addition to ramping up faster to reach peak throughput.

5 TRANSPORT SELECTION

Throughput profiles generated from the measurements can be used to select a configuration (V, n, B) based on RTT τ to achieve peak throughput as described next in Section 5.1. Furthermore, the corresponding throughput estimate $\hat{\Theta}_O(\tau)$ for this chosen configuration will be close to the actual peak throughput in a statistical sense with a high probability (as will be shown in Section 5.2).

5.1 Selection of Transport

Throughput profiles are generated by codes that sweep the parameters (V, n, B), and can be used as follows:

1. Determine RTT to destination using ping.
2. Use throughput profiles to determine a TCP variant and its parameters with the highest throughput if measurements are available at that RTT or by linearly interpolating the measurements otherwise.

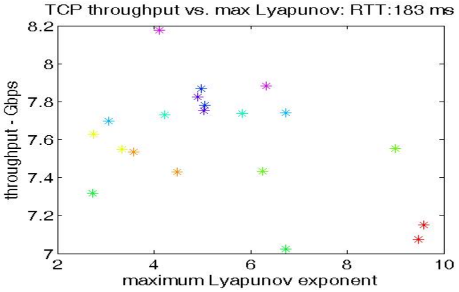


Figure 14: Average throughput vs. Lyapunov exponent for 10-stream CUBIC with 183 **ms RTT SONET and large buffers**

3. Load the congestion control module into kernel and set up the parameters.

Based on our measurements, this procedure selects STCP with multiple streams for smaller RTTs, which provides higher throughput compared to CUBIC, the Linux default.

5.2 Confidence Estimates

The throughput $\theta(\tau, t)$ is a random quantity whose distribution $\mathbf{P}_{\Theta_O(\tau)}$ is quite complex since it depends on the congestion control mechanism, and dynamics of the connection and host. We define the *profile regression* as

$$\bar{\Theta}_O(\tau) = E\left[\Theta_O(\tau)\right] = \int \Theta_O(\tau) d\mathbf{P}_{\Theta_O(\tau)},$$

which can be estimated based on measurements $\theta(\tau_k, t_i^k)$ at τ_k, $k = 1, 2, \ldots, n$, and times t_i^k, $i = 1, 2, \ldots, n_k$. It exhibits an overall decreasing trend with a concave region for $\tau \in [0, \tau_T]$ followed by a convex region for $\tau > \tau_T$. The throughput estimate, given by the *profile mean* $\hat{\Theta}_O(\tau)$, is computed using measurements as

$$\hat{\Theta}_O(\tau_k) = \frac{1}{n_k}\sum_{i=1}^{n_k}\theta(\tau_k, t_i^k)$$

at τ_k's and linearly interpolated between them. Note that $\hat{\Theta}_O(\tau)$, computed entirely using measurements, is indicative of the actual throughput at RTT τ, whose expected value is $\bar{\Theta}_O(\tau)$. We will now show it is indeed a good estimate of $\bar{\Theta}_O(\tau)$, in terms of expected error, and furthermore its performance improves with more measurements, independent of the underlying distribution $\mathbf{P}_{\Theta_O(\tau)}$.

Consider an estimate $f(.)$ of $\bar{\Theta}_O(.)$ based on measurements from a function class $\mathcal{M}$ of unimodal functions, which includes the dual-regime monotone throughput profiles as a special case. The *expected error* $I(f)$ of the estimator f is $I(f) = \int [f(\tau) - \theta(\tau, t)]^2 d\mathbf{P}_{\theta(\tau,t)}$, and the *best estimator* f^* is given by $\hat{I}(f^*) = \min_{f \in \mathcal{M}} I(f)$. The *best empirical estimator* $\hat{f} \in \mathcal{M}$ minimizes the empirical error

$$\hat{I}(f) = \frac{1}{n}\sum_{k=1}^{n}\frac{1}{n_k}\sum_{j=1}^{n_k}\left[f(\tau_k) - \theta(\tau_k, t_j)\right]^2,$$

that is, $\hat{I}(\hat{f}) = \min_{f \in \mathcal{M}} \hat{I}(f)$. Since $\hat{\Theta}_O(\tau)$ is the response mean at each RTT τ_k, it achieves the minimum empirical error. By using Vapnik-Chervonenkis theory [29], we have

$$\begin{aligned}
&\mathbf{P}\left\{I\left(\hat{\Theta}_O\right) - I(f^*) > \epsilon\right\} \\
&\quad\leq \mathbf{P}\left\{\max_{h \in \mathcal{M}} \left|I_D(h) - \hat{I}_D(h)\right| > \epsilon/2\right\} \\
&\quad\leq 16\mathcal{N}_\infty\left(\frac{\epsilon}{C}, \mathcal{M}\right) n e^{-\epsilon^2 n/(4C)^2}
\end{aligned}$$

where $\theta(\tau, t) \leq C$, and $\mathcal{N}_\infty(\epsilon, \mathcal{M})$ is the ϵ-cover of $\mathcal{M}$ under L_∞ norm. Due to the unimodality of functions in $\mathcal{M}$, their total variation is upper-bounded by $2C$, which provides us the upper bound ([4], p. 175):

$$\mathcal{N}_\infty\left(\frac{\epsilon}{C}, \mathcal{M}\right) < 2\left(\frac{n}{\epsilon^2}\right)^{(1+C/\epsilon)\log_2(2\epsilon/C)}.$$

By using this bound, we obtain

$$\mathbf{P}\left\{I\left(\hat{\Theta}_O\right) - I(f^*) > \epsilon\right\}$$
$$< 32\left(\frac{n}{\epsilon^2}\right)^{(1+C/\epsilon)\log_2(4\epsilon/C)} n e^{-\epsilon^2 n/(2C)^2}.$$

The exponential term on the right-hand side decays faster in n than other terms, hence for sufficiently large n it would be smaller than a given probability α.

In summary, the expected error $I(\hat{\Theta}_O)$ of the response mean is within ϵ of the optimal error $I(f^*)$ with a probability that increases with the number of observations. This performance guarantee is independent of how complex the underlying distribution $\mathbf{P}_{\Theta_O(\tau)}$ is. Thus, $\hat{\Theta}_O(\tau)$ is a good estimate of the actual peak throughput achievable at RTT τ independent of the underlying distribution, which is a complex composition of the effects of host systems and connection hardware as well as TCP/IP stack.

6 CONCLUSIONS

Wide-area data transfers in HPC infrastructures are increasingly being carried over dedicated network connections, driven in part by the expectation of high throughput and stable dynamics. In many cases, the underlying transport is provided by TCP for memory and file transfers, but its analyses and measurements over dedicated connections are limited, making it harder to assess its impact on application performance. To study the performance of TCP variants and their parameters for high-performance transfers over dedicated connections, we collected systematic measurements using physical and emulated dedicated connections. They revealed important properties such as concave regions and relationships between dynamics and throughput profiles. Interestingly, the dynamics are much richer than expected, as revealed by the Poincaré map and Lyapunov exponent estimates. We presented analytical results that identify RTT ranges corresponding to concave and high throughput profiles. The measurements and analyses enable the selection of a high throughput transport method and corresponding parameters for a given connection based on RTT.

Future directions include more detailed analytical models that closely match the measurements under packet drops and other errors with variable file and disk I/O capacities, particularly when they significantly impact TCP throughput dynamics. Also of future interest are enhancements of the current first-principle TCP models to explain the dual-mode throughput profiles by integrating dynamics parameters such as the Lyapunov exponents, and incorporation of throughput profiles into SDN technologies to select and set up suitable paths to match the transport protocols.

ACKNOWLEDGMENTS

This work is funded by RAMSES project and the Applied Mathematics Program, Office of Advanced Computing Research, U.S. Department of Energy, and by Extreme Scale Systems Center, sponsored by U. S. Department of Defense, and performed at Oak Ridge National Laboratory managed by UT-Battelle, LLC for U.S. Department of Energy under Contract No. DE-AC05-00OR22725.

REFERENCES

[1] Bay microsystems. http://www.baymicrosystems.com. Accessed: Apr. 2017.

[2] A. Aguilera, M. Kluge, , T. William, and W. E. Nagel. *HPC File Systems in Wide Area Networks: Understanding the Performance of Lustre over WAN*, pages 65–76. Springer Berlin Heidelberg, 2012.

[3] K. T. Alligood, T. D. Sauer, and J. A. Yorke. *Chaos: An Introduction to Dynamical Systems*. Springer-Verlag Pub., Reading, MA, 1996.

[4] M. Anthony and P. L. Bartlett. *Neural Network Learning: Theoretical Foundations*. Cambridge University Press, 1999.

[5] S. Atchley, D. Dillow, G. Shipman, P. Geoffray, J. M. Squyres, G. Bosilca, and R. Minnich. The common communication interface (CCI). In *19th IEEE Symposium on High Performance Interconnects (HOTI), Santa Clara, CA*, 2011.

[6] M. Avriel, W. E. Diewert, S. Schaible, and I. Zang. *Generalized Concaviy*. SIAM, 2010.

[7] Science DMZ: Data Transfer Nodes. https://fasterdata.es.net/science-dmz/DTN. Accessed: Apr. 2017.

[8] Energy Sciences Network. http://www.es.net. Accessed: Apr. 2017.

[9] Y. Gu and R. L. Grossman. UDT: UDP-based data transfer for high-speed wide area networks. *Computer Networks*, 51(7), 2007.

[10] M. Hassan and R. Jain. *High Performance TCP/IP Networking: Concepts, Issues, and Solutions*. Prentice Hall, 2004.

[11] S. Jain, A. Kumar, S. Mandal, et al. B4: Experience with a globally-deployed software defined WAN. *SIGCOMM Comput. Commun. Rev.*, 43(4):3–14, Oct. 2013.

[12] T. Kelly. Scalable TCP: Improving performance in high speed wide area networks. *Computer Communication Review*, 33(2):83–91, 2003.

[13] Q. Liu, J. Logan, Y. Tian, H. Abbasi, N. Podhorszki, J. Y. Choi, et al. Hello ADIOS: the challenges and lessons of developing leadership class I/O frameworks. *Concurrency and Computation: Practice and Experience*, 26(7):1453–1473, 2014.

[14] Q. Liu, N. S. V. Rao, C. Q. Wu, D. Yun, R. Kettimuthu, and I. Foster. Measurements-based analysis of performance profiles and dynamics of udp transport protocols. In *International Conference on Network Protocols*. 2016.

[15] M. Mathis, J. Semke, J. Mahdavi, and T. Ott. The mascroscopic behavior of the TCP congestion avoidance algorithm. *Computer Communication Review*, 27(3), 1997.

[16] Obsidian Research Corporation, http://www.obsidianresearch.com. Accessed: Apr. 2017.

[17] On-demand secure circuits and advance reservation system. http://www.es.net/oscars. Accessed: Apr. 2017.

[18] J. Padhye, V. Firoiu, D. F. Towsley, and J. F. Kurose. Modeling TCP Reno performance: A simple model and its empirical validation. *IEEE/ACM Transactions on Networking*, 8(2):133–145, 2000.

[19] S. Parete-Koon, B. Caldwell, S. Canony, E. Dartz, J. Hicky, J. Hill, et al. HPC's pivot to data. In *Cray User's Group Meeting*, 2014.

[20] N. S. V. Rao, J. Gao, and L. O. Chua. On dynamics of transport protocols in wide-area internet connections. In *Complex Dynamics in Communication Networks*. Springer-Verlag Publishers, 2005.

[21] N. S. V. Rao, Q. Liu, S. Sen, G. Hinkel, N. Imam, I. Foster, R. Kettimuthu, C. Q. Wu, and D. Yun. Experimental analysis of file transfer rates over wide-area dedicated connections. In *18th IEEE International Conference on High Performance Computing and Communications (HPCC)*, Dec. 2016.

[22] N. S. V. Rao, D. Towsley, G. Vardoyan, B. W. Settlemyer, I. T. Foster, and R. Kettimuthu. Sustained wide-area TCP memory transfers over dedicated connections. In *IEEE International Conference on High Performance and Smart Computing*, 2015.

[23] N. S. V. Rao, W. Yu, W. R. Wing, S. W. Poole, and J. S. Vetter. Wide-area performance profiling of 10GigE and InfiniBand technologies. In *Supercomputing Conference*, 2008.

[24] I. Rhee and L. Xu. CUBIC: A new TCP-friendly high-speed TCP variant. In *Proceedings of the Third International Workshop on Protocols for Fast Long-Distance Networks*, 2005.

[25] T. Shanley. *InfiniBand Network Architecture*, volume I and II. MindShare, Inc., 2003.

[26] R. Shorten and D. Leith. H-TCP: TCP for high-speed and long-distance networks. In *Proceedings of the Third International Workshop on Protocols for Fast Long-Distance Networks*, 2004.

[27] Y. Srikant and L. Ying. *Communication Networks: An Optimization, Control, and Stochastic Networks Perspective*. Cambridge University Press, 2014.

[28] GT 4.0 GridFTP. http://www.globus.org. Accessed: Apr. 2017.

[29] V. N. Vapnik. *Statistical Learning Theory*. John-Wiley and Sons, New York, 1998.

[30] XDD - The eXtreme dd toolset, https://github.com/bws/xdd. Accessed: Apr. 2017.

[31] T. Yee, D. Leith, and R. Shorten. Experimental evaluation of high-speed congestion control protocols. *Transactions on Networking*, 15(5):1109–1122, 2007.